



VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ

BRNO UNIVERSITY OF TECHNOLOGY

FAKULTA INFORMAČNÍCH TECHNOLOGIÍ

FACULTY OF INFORMATION TECHNOLOGY

ÚSTAV POČÍTAČOVÉ GRAFIKY A MULTIMÉDIÍ

DEPARTMENT OF COMPUTER GRAPHICS AND MULTIMEDIA

NÁSTROJ PRO SLEDOVÁNÍ POSTUPU NA ZÁLEŽI- TOSTECH

ONLINE TOOL FOR TRACKING PROGRESS ON ISSUES

BAKALÁŘSKÁ PRÁCE

BACHELOR'S THESIS

AUTOR PRÁCE

AUTHOR

SILVIA JANÍKOVÁ

VEDOUCÍ PRÁCE

SUPERVISOR

Prof. Ing. ADAM HEROUT, Ph.D.

BRNO 2017

Vysoké učení technické v Brně - Fakulta informačních technologií

Ústav počítačové grafiky a multimédií

Akademický rok 2016/2017

Zadání bakalářské práce

Řešitel: **Janíková Silvia**

Obor: Informační technologie

Téma: **Nástroj pro sledování postupu na záležitostech
Online Tool for Tracking Progress on Issues**

Kategorie: Uživatelská rozhraní

Pokyny:

1. Seznamte se s existujícími jednoduchými nástroji/aplikacemi pro zvyšování produktivity a monitorování pokroku.
2. Navrhněte jednoduchou webovou aplikaci pro sledování postupu na záležitostech. Zaměřte se na jednoduchost sdílení a uživatelskou zkušenost.
3. Prototypujte dílčí varianty vytvářeného uživatelského rozhraní a testujte návrhy na uživateli.
4. Integrujte dílčí prvky uživatelského rozhraní do funkční webové aplikace a testujte ji na uživateli.
5. Zhodnoťte dosažené výsledky a navrhněte možnosti pokračování projektu; vytvořte plakátek a krátké video pro prezentování projektu.

Literatura:

- dle pokynů vedoucího

Podrobné závazné pokyny pro vypracování bakalářské práce naleznete na adrese

<http://www.fit.vutbr.cz/info/szz/>

Technická zpráva bakalářské práce musí obsahovat formulaci cíle, charakteristiku současného stavu, teoretická a odborná východiska řešených problémů a specifikaci etap (20 až 30% celkového rozsahu technické zprávy).

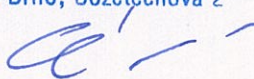
Student odevzdá v jednom výtisku technickou zprávu a v elektronické podobě zdrojový text technické zprávy, úplnou programovou dokumentaci a zdrojové texty programů. Informace v elektronické podobě budou uloženy na standardním nepřepisovatelném paměťovém médiu (CD-R, DVD-R, apod.), které bude vloženo do písemné zprávy tak, aby nemohlo dojít k jeho ztrátě při běžné manipulaci.

Vedoucí: **Herout Adam, prof. Ing., Ph.D., UPGM FIT VUT**

Datum zadání: 1. listopadu 2016

Datum odevzdání: 17. května 2017

VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ
Fakulta informačních technologií
Ústav počítačové grafiky a multimédií
612 00 Brno, Božetěchova 2



doc. Dr. Ing. Jan Černocký
vedoucí ústavu

Abstrakt

Táto práca sa zaoberá návrhom a tvorbou webovej aplikácie. Cieľom tejto aplikácie bude sledovanie postupu práce v rôznych oblastiach s možným zvyšovaním produktivity. Práca sa zaoberá možnosťami tvorby webových aplikácií, návrhom užívateľského rozhrania aplikácie a jej implementáciou pomocou frameworku Django. V praxi by mala aplikácia umožniť užívateľom efektívne sledovanie pokroku vrámci skupiny spolupracovníkov.

Abstract

This thesis focuses on the design and development of the web application. The aim of this application is to track progress in various fields, which can cause productivity increasing. This thesis also describes approaches to web applications development, user interface design and implementation of this application using Django framework. In practise the application should enable its users to effectively track progress within a group of co-workers.

Klíčové slová

užívateľské rozhranie, webová aplikácia, Django, sledovanie pokroku

Keywords

user interface, web application, Django, tracking progress

Citácia

JANÍKOVÁ, Silvia. *Nástroj pro sledování postupu na záležitostech*. Brno, 2017. Bakalářská práce. Vysoké učení technické v Brně, Fakulta informačních technologií. Vedoucí práce Herout Adam.

Nástroj pro sledování postupu na záležitostech

Prehlásenie

Prehlasujem, že som túto bakalársku prácu vypracovala samostatne pod vedením pána prof. Ing. Adama Herouta, Ph.D. Uviedla som všetky literárne pramene a publikácie, z ktorých som čerpala.

.....
Silvia Janíková
17. mája 2017

Podakovanie

Chcela by som poďakovať svojmu vedúcemu prof. Ing. Adamovi Heroutovi, Ph.D. za jeho odborné rady, vedomosti a trpezlivosť pri riešení tejto bakalárskej práce.

Obsah

1 Úvod	2
2 Monitorovanie produktivity	3
2.1 Motivácia	3
2.2 Existujúce aplikácie a nástroje	3
3 Štúdium a použité technológie	6
3.1 Princípy návrhu webových aplikácií	6
3.2 Použité technológie	8
4 Návrh aplikácie na sledovanie postupu	11
4.1 Cieľová skupina	11
4.2 Požiadavky na výsledný systém	12
4.3 Návrh užívateľského rozhrania	12
4.4 Návrh serverovej časti	16
5 Implementácia	18
5.1 Grafické rozhranie systému	18
5.2 Komunikácia medzi klientskou a serverovou časťou	19
5.3 Nasadenie aplikácie na server	21
5.4 Zdieľanie prostredníctvom URL	21
5.5 Autentifikácia pomocou Google API	21
6 Testovanie a možnosti ďalšieho rozvoja aplikácie	22
6.1 Testovanie	22
6.2 Výsledky testovania	22
6.3 Možnosti ďalšieho rozvoja aplikácie	23
7 Záver	25
Literatúra	26
Prílohy	27
A Obsah priloženého CD	28

Kapitola 1

Úvod

V súčasnosti sa webové aplikácie využívajú v mnohých odvetviach. Využívanie webových aplikácií pri práci je populárne hlavne kvôli jednoduchosti užívateľského rozhrania, prenositeľnosti medzi platformami a dostupnosti. Existujú spoločnosti, ktoré pri organizácii práce uprednostňujú webové nástroje pred zložitými systémami. Cieľom tejto práce je navrhnúť a vytvoriť systém, ktorý by odľahčil komunikáciu súvisiacu s predávaním informácií o pokroku práce.

Základnou požiadavkou na výsledný systém je jednoduchosť z užívateľského hľadiska a rýchle šírenie medzi užívateľmi. Systém bude poskytovať funkcie na podávanie informácií o aktuálnom vývoji práce a v neposlednom rade motivovať užívateľa k vyššej produktivite.

V nasledovnej kapitole je bližšie špecifikovaná motivácia na vytvorenie aplikácie na sledovanie postupu a podrobne rozobraté existujúce systémy s podobným zameraním. Ďalej sa v práci zaoberám procesom návrhu webových aplikácií a technológiami, ktoré sa na to využívajú. Druhá časť práce je venovaná návrhu, implementácii a testovaniu výslednej aplikácie.

Kapitola 2

Monitorovanie produktivity

V tejto kapitole je objasnená motivácia k tvorbe nástroja na sledovanie postupu a existujúce riešenia s podobným zameraním. V súčasnosti existuje niekoľko aplikácií, ktoré sa zameriavajú na plánovanie, zvyšovanie produktivity a oblasť manažmentu. V nasledujúcich odstavcoch bude bližšie predstavená aplikácia Trello, ktorá sa s obľubou používa na manažovanie projektov a zadaní. Z aplikácií s podobným zadaním bude bližšie opísaná aj Asana, ktorá slúži na komplexnejšiu správu a organizáciu pracovných úloh a komunikačný nástroj Glip.

2.1 Motivácia

Pri práci v tíme sa často naráža na problém, ako komunikovať tak, aby sa zosúlادili informácie potrebné k efektívnemu napredovaniu. Komunikácia a referovanie stavu projektu sa tak realizuje prostredníctvom emailov alebo iných komunikačných nástrojov. Tento spôsob vyhodnocovania aktuálneho stavu projektu je však často zdĺhavý alebo neefektívny. Prekážkou k vyššej produktivite býva napríklad nejasné stanovenie cieľov, alebo neobjektívne sebahodnotenie.

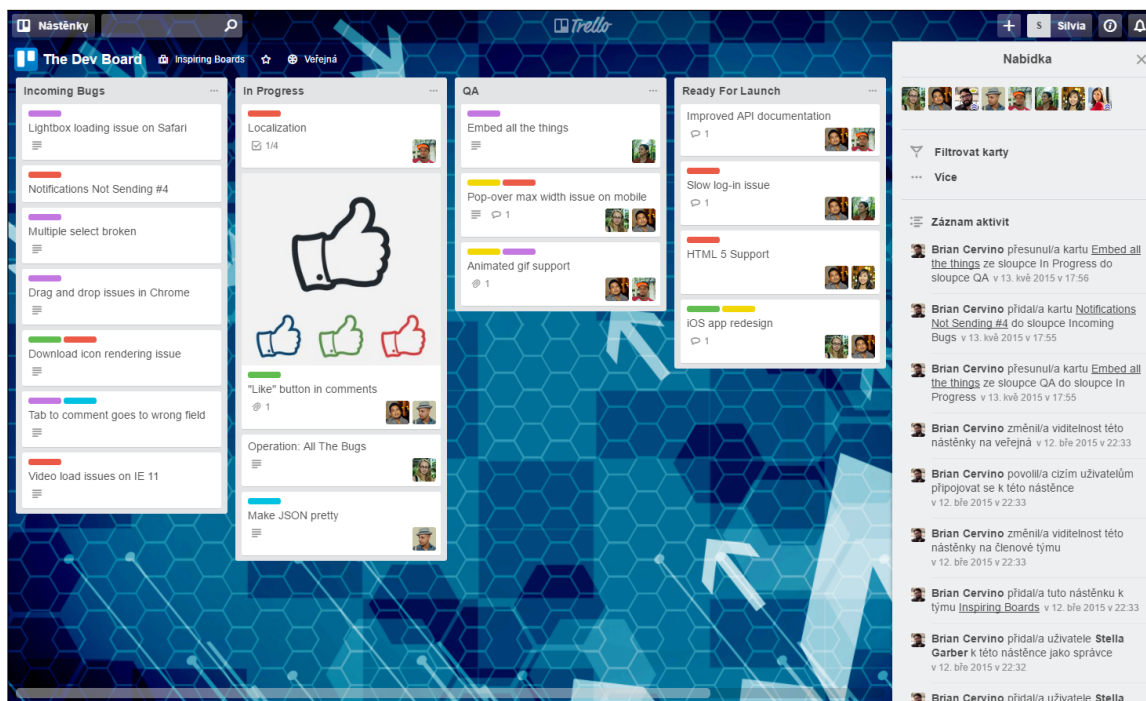
Tento projekt sa zameriava na odbúranie zbytočnej réžie súvisiacej s predávaním informácií o pokroku práce. Jeho hlavnou výhodou je názornosť a jednoduchosť používania. Slúži na mapovanie aktuálnej situácie z akýchkoľvek oblastí, v ktorých je možné zaznamenávať progres na jednoduchý graf, vyjadrujúci mieru splnenej práce. Tým pádom je použiteľný pre široké spektrum užívateľov.

2.2 Existujúce aplikácie a nástroje

Na uľahčenie organizácie a manažmentu projektov sa čoraz viac používajú jednoduché webové nástroje. U vybraných aplikácií bude priblížený princíp fungovania a užívateľská skúsenosť pri práci s nimi.

2.2.1 Trello

Zrejme najpoužívanejšou aplikáciou na správu projektov je Trello [3]. Trello je webová aplikácia, ktorá na organizáciu práce využíva koncept Kanban. Princíp Kanbanu spočíva v rozdelení tabule na stĺpce odpovedajúce etapám úlohy projektu a jednotlivé úlohy sa potom presúvajú medzi stĺpcami smerom doprava. Najľavejší stĺpec na tabuli predstavuje počiatočný stupeň vývoja a najpravejší stĺpec predstavuje finálnu fázu.



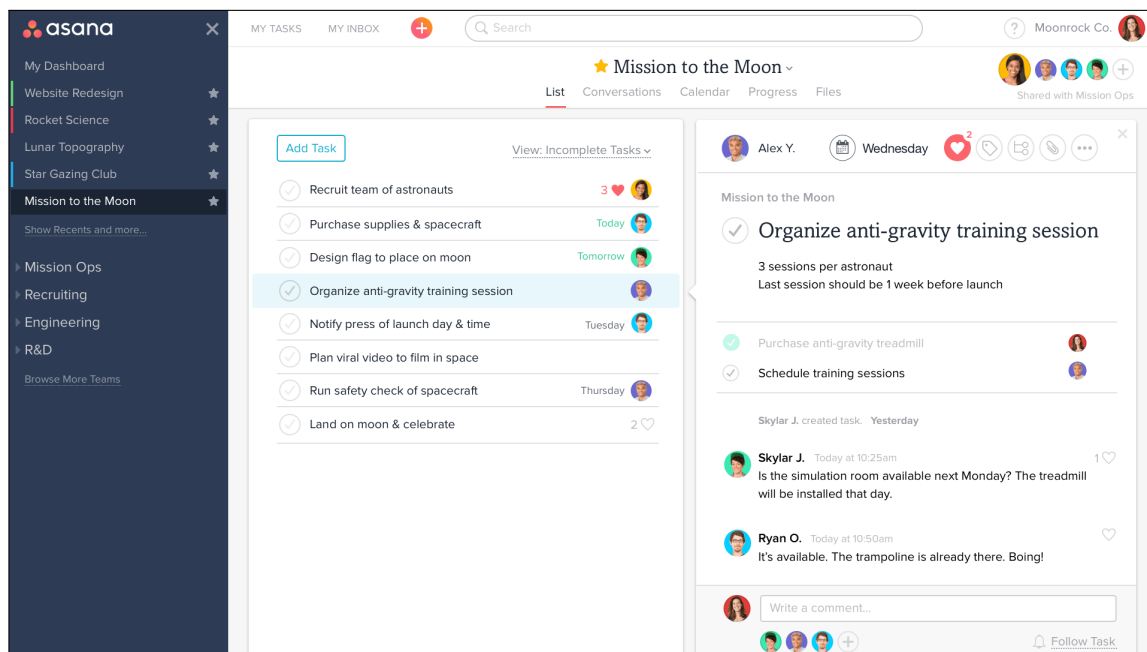
Obr. 2.1: Užívateľské rozhranie aplikácie Trello tvorené stĺpcami, ktoré charakterizujú systém Kanban.

Užívateľské rozhranie, ako je vidieť na obrázku 2.1, tvoria tri základné časti: nástenky, zoznamy a karty. Nástenka zoskupuje konkrétny projekt, produkt alebo akúkoľvek organizačnú štruktúru. Každá nástenka je zložená zo zoznamov, ktoré sa zobrazujú ako stĺpce. Karty môžu predstavovať jednotlivé úlohy alebo poznámky a je možné ich kategorizovať ťahaním medzi stĺpcami. Hlavnou výhodou tejto aplikácie je vizuálne zobrazenie prehľadu nad prebiehajúcimi projektmi. Veľkým plusom je intuitívne a hravé užívateľské rozhranie a flexibilita použitia. Trello je dobrým príkladom toho, aký potenciál majú jednoduché aplikácie a dokážu prilákať aj veľkú skupinu zákazníkov, čím dokážu byť konkurencieschopný [13].

2.2.2 Asana

Medzi populárne aplikácie na správu projektov sa radí aj Asana [4]. Umožňuje vytvárať štruktúru projektu pridávaním cieľov a mílnikov. Ku každej úlohe je tiež možné pridať linky, poznámky a komentáre, ktoré sú viditeľné v rámci celej pracovnej skupiny.

Z obrázku 2.2 je zrejmé, že užívateľské rozhranie aplikácie sa skladá z ľavého panelu, ktorý obsahuje projekty, tímy a pracovné prostredia, hlavného okna v ktorom sa zobrazuje obsah na základe vybranej položky, a pravého informačného bloku, v ktorom sa menia informácie na základe obsahu hlavného okna. Asana umožňuje spravovať termíny projektov, ktoré je možné zároveň synchronizovať s kalendármi iných služieb. Táto aplikácia poskytuje množstvo funkcií, ktoré môžu byť užitočné hlavne pri manažovaní väčších projektov, avšak pre bežného užívateľa môže byť práca s týmto nástrojom komplikovaná.



Obr. 2.2: Základná obrazovka aplikácie Asana, na ktorej je vidieť rozdelenie aplikácie na hlavný obsah, dodatočné informácie a panely s nástrojmi.

2.2.3 Glip

Medzi aplikácie využívané pri tímovej spolupráci sa radí aj aplikácia Glip [12]. Glip je komunikačný nástroj, ktorý okrem bežnej komunikácie poskytuje aj rozširujúce funkcie ako napríklad manažovanie úloh, video konferencie alebo zdieľanie kalendára. Táto aplikácia umožňuje synchronizáciu s emailovým klientom, vďaka čomu je možné komunikovať aj s ľuďmi, ktorí aplikáciu priamo nevyužívajú.

Veľkým plusom tejto aplikácie je jednoduché a prehľadné užívateľské rozhranie, ktorého dominantnú časť tvorí blok s konverzáciou. Prostredníctvom položiek ľavého panelu je možné zobrazovať obľúbené konverzácie, ľudí v rámci pracovnej skupiny, dashboard alebo kalendár s úlohami.

Kapitola 3

Štúdium a použité technológie

Prvá časť tejto kapitoly je venovaná princípom návrhu webových aplikácií. Obsahuje popis jednotlivých fáz návrhu a sumarizuje zásady tvorby užívateľského rozhrania. V druhej časti tejto kapitoly sú bližšie opísané technológie, ktoré sa v súčasnosti využívajú pri tvorbe webových aplikácií.

3.1 Princípy návrhu webových aplikácií

Existujú rôzne prístupy k tvorbe návrhu webových aplikácií. V nasledujúcej časti budú vysvetlené jednotlivé fázy návrhu bez ohľadu na konkrétny manažérsky prístup.

3.1.1 Proces návrhu webu

Podľa Jana Řezáče [14] proces návrhu webu zahrňuje fázy: objavovanie, užívateľský výskum, návrh webu a evaluácia. Fáza **objavovanie** je zameraná na vytvorenie predstavy o koncepte a zmysle výslednej webovej aplikácie. Výsledkom tejto časti by mala byť predstava o tom, komu bude výsledný produkt určený a prečo by užívateľ mal používať výslednú službu. V rámci tejto fázy sa realizuje aj tzv. *branding*. Branding je činnosť prostredníctvom ktorej dostáva značka zmysel. Brand odlišuje jednu značku od druhej, má typicky aj vizuálnu reprezentáciu, ale to neznamená, že brand je tvorený iba logom.

Neoddeliteľnou súčasťou návrhu webu je **užívateľský výskum**. Cieľom užívateľského výskumu je minimalizovať riziko vytvorenia nezmysluplného výstupu, zároveň sa vďaka nemu stanoví priority celého projektu z pohľadu návštevníkov webu. Metódy užívateľského výskumu je možné realizovať online alebo v teréne. Medzi online výskumné metódy patrí analýza súčasného webu, dotazníkový prieskum, analýza webu konkurencie a analýza kľúčových slov. Výraznou výhodou online metód je zhromaždenie užitočných faktov a štatistík. Výskum v teréne je naopak zameraný na užívateľa. Medzi takéto metódy patrí *Card Sorting*, ktorého cieľom je pochopiť ako ľudia uvažujú o pojmoch a ich kategóriách, čo je užitočné pri návrhu informačnej architektúry webu. Ďalšou metódou realizovanou v teréne sú hĺbkové rozhovory s ľuďmi, ktorí sú vhodní pre konkrétny projekt. Z týchto rozhovorov je možné získať podnety pre informačnú architektúru a pochopiť jazyk užívateľa a jeho skutočné problémy. Podobnou metódou je tieňovanie, ktoré spočíva v tom, že vývojár sa doslova stáva tieňom skúmaného človeka, napríklad počas jeho pracovnej doby. Táto metóda býva užitočná najmä pri tvorbe informačného systému.

Prvým krokom pri **návrhu webu** je stanovenie smeru, čiže interpretácia myšlienok z predchádzajúcich dvoch fáz do predstavy o webe. Po vytvorení štruktúry a obsahu webu

nasleduje prototypovanie. V rámci prototypu je navrhnuté rozloženie obsahu webu a jeho priority na jednotlivých stránkach. Po previazaní stránok je možné prechádzať kľúčové interakcie návštevníka s webom a výstup otestovať. Pri tvorbe prototypu sa od skicovania, pri ktorom sa prevádza obsah do hrubej vizuálnej podoby, prechádza k *wireframom*¹, ktoré tvoria základ prototypu. Posledným krokom návrhu je grafický návrh. Kvalitná grafika podporuje pozíciu značky, zvyšuje dôveryhodnosť webu a tolerantnosť návštevníkov k chybám.

Poslednou fázou procesu návrhu webových aplikácií je **evaluácia**. Cieľom tejto fázy je vylepšiť aktuálne riešenie na základe spätnej väzby. Evaluáciu je možné realizovať užívateľským testovaním. Užívateľské testovanie môže prebiehať v rôznych formách, najčastejšie sa používa kvalitatívne užívateľské testovanie od Kruga. Tento typ testovania sa realizuje výberom 3–5 respondentov, ktorí plnia konkrétne úlohy vo vytvorenom prototypu, pričom sú zaznamenávané ich reakcie. Spätnú väzbu je možné získať aj využitím checklistu, čiže zoznamu bodov, ktoré je možné na webe overiť.

3.1.2 Zásady tvorby užívateľského rozhrania

Steve Krug vo svojej knihe [2] uvádza, že základnou zásadou tvorby užívateľských rozhraní je nenútiť užívateľa premýšľať. Znamená to, že webová aplikácia by mala byť na prvý pohľad intuitívna, pochopiteľná a samovysvetľujúca. Malo by byť zrejmé, aký zmysel má aplikácia a ako ju používať, bez toho aby o tom musel užívateľ zvlášť premýšľať.

Ľudia trávajú veľmi krátku dobu čítaním textov na webových stránkach. Namiesto čítania ich užívatelia len prehliadajú a pátrajú po slovách, ktoré upútajú ich pozornosť. Ak má byť stránka pochopiteľná, je dôležité, aby vzhľad vecí na stránke – všetky vizuálne podnety – jasne a presne zobrazovali vzťah medzi objektmi na stránke. Jasnú vizuálnu hierarchiu je možné dosiahnuť zvýrazňovaním dôležitých elementov, vizuálnym prepojením objektov, ktoré spolu logicky súvisia a vizuálnym zanorovaním objektov, ktoré sú niečoho súčasťou.

Weboví návrhári používajú termín perzistentná navigácia, ktorým popisujú sadu navigačných prvkov, ktoré sa objavujú na každej stránke serveru. Prítomnosť navigácie na každej stránke a jej konzistentný vzhľad dáva užívateľom istotu, že sú stále na rovnakom serveri. Perzistentná navigácia by mala obsahovať päť prvkov: logo serveru, odkaz na domovskú stránku, vyhľadávanie, pomôcky a sekcie. Pokiaľ vezmeme do úvahy počet užívateľov, ktorý k orientácii na stránke používajú vyhľadávanie, tak by mala každá stránka obsahovať vyhľadávacie pole, alebo odkaz na stránku s vyhľadávaním.

Pri tvorbe užívateľského rozhrania je potrebné zohľadniť veci, ktoré znižujú priazeň užívateľov:

- **Skrývanie informácií** potrebných pre užívateľa. Najčastejšie sú skryté telefónne čísla zákazníckej podpory, poplatky za dopravu a ceny.
- **Premýšľanie** o tom ako formátovať zadané údaje. Užívateľ by nikdy nemal premýšľať o tom v akom formáte by mal zadávať údaje – napríklad ako či je potrebné vyplniť lomku v rodnom čísle alebo medzery v čísle kreditnej karty.
- **Požadovanie nepotrebných informácií**. Mnoho užívateľov je skeptických voči požiadavkám na osobné údaje.
- **Kladenie prekážok** do cesty. Užívatelia sa potrebujú dostať čo najrýchlejšie k relevantným informáciám. Je veľmi neprijemné pokiaľ ich v tom spomaľujú dlhé úvody, alebo stránky preplnené marketingovými fotografiami.

¹náčrt rozloženia prvkov a obsahu aplikácie

- **Amatérsky pôsobiace stránky.** Priazeň užívateľov je možné stratiť, ak stránky vyzerajú nedokončene, neusporiadane alebo neprofesionálne, ako keby ich vzhľadu nikto nevenoval dostatok pozornosti.

Aj keď webová aplikácia obsahuje niektoré z predchádzajúcich chýb, je možné zvýšiť priazeň užívateľov zaradením vecí, ktoré kladú dôraz na ich záujmy:

- **Ozrejmenie a zjednodušenie základných vecí,** ktoré chcú užívatelia na stránke vykonávať.
- **Poskytovanie informácií, ktoré užívateľ potrebuje vedieť.** Užívateľ by mal byť informovaný o dodatočných nákladoch na dopravu, parkovacích poplatkoch alebo nefungujúcich službách, aj keď sa jedná o informácie, ktoré firmy neradi uvádzajú.
- **Minimalizovanie nevyhnutých krokov.**
- **Prispôsobenie stránok k tlači.** Každý užívateľ určite ocení odstránenie prebytočných reklám a nepotrebných informácií pri tlačení informácií zo stránok.
- **Zjednodušenie zotavenia z chýb.** Absolvovaním užívateľského testovania sa predchádza mnohým chýbám, na ktoré by mohol užívateľ naraziť. Napriek tomu, tam kde hrozí riziko potencionálnych chýb, je potrebné poskytnúť elegantný spôsob, ako sa s nimi vysporiadať.

3.2 Použité technológie

Základom vizuálu každého webu je HTML (*HyperText Markup Language*), CSS (*Cascading Style Sheets*) a Javascript. Tieto technológie sú často dopĺňané frameworkmi, ktorých úlohou je urýchliť vývoj. Pri tvorbe klientskej časti aplikácie boli použité: Bootstrap, jQuery a jQuery UI.

Vývoj serverovej časti aplikácie je možné realizovať prostredníctvom niektorého z webových frameworkov. Cieľom webových frameworkov je automatizovať niektoré časti vývoja webových aplikácií ako napríklad prácu s databázou, vykresľovanie stránok alebo správu sedení.

3.2.1 Bootstrap

Bootstrap je framework [11], ktorý je zameraný na vytváranie responzívnych webových aplikácií. Je tvorený kombináciou HTML, CSS a Javascriptu a umožňuje tvorbu rozsiahlejších stránok jednoduchším a rýchlejším spôsobom. Okrem podpory responzivity poskytuje rôzne predpripravené komponenty ako modálne okná, formuláre alebo tlačítka. Užitočný je aj tzv. *grid system*, ktorý umožňuje rozmiestniť elementy v rámci mriežky, ktorá sa prispôbuje veľkosti zobrazenia konkrétneho zariadenia.

3.2.2 jQuery a jQuery UI

Knižnica jQuery [9] vytvára abstraktnú vrstvu klasického skriptovania. Poskytuje funkcie, ktoré sa bežne realizujú prostredníctvom Javascriptu. Umožňuje prístup k prvkom v dokumente, zmenu vzhľadu aplikácie (pridaním CSS tried alebo zmenou individuálneho štýlu), zmenu obsahu dokumentu, animáciu vykonaných zmien a získavanie informácií zo serveru bez nutnosti obnovenia stránky. jQuery UI je sada interagujúcich prvkov, efektov, ovládacích prvkov a tém postavených na jQuery, CSS a HTML.

3.2.3 Django

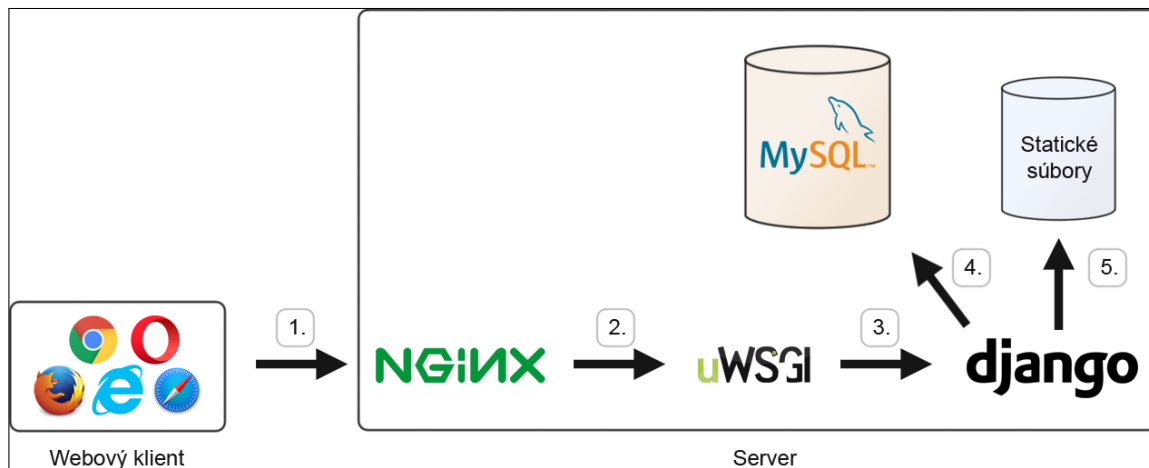
Na vývoj serverovej časti aplikácie bol zvolený webový framework Django [5]. Tento framework je napísaný v jazyku Python a je postavený na základoch návrhového vzoru model-view-controller [1], ktorý oddeľuje logické časti aplikácie. Hlavnou komponentou tohto návrhového vzoru je model. Predstavuje abstraktnú reprezentáciu dát aplikácie. Jednotlivých modelov môže byť v aplikácii viac a reprezentujú databázové tabuľky relačnej databáze. Django zabezpečuje možnosť výberu konkrétnej relačnej databázy podľa potreby aplikácie. View je prezentačná vrstva modelu, to znamená, že je možné dynamicky generovať jednotlivé HTML dokumenty podľa požiadaviek klienta. Tieto HTML dokumenty sú uložené na strane serveru v podobe šablón. Pomocou premenných, podmienok a cyklov sa vygeneruje výsledný obsah webovej aplikácie. Tieto konštrukcie sa vkladajú do šablón pomocou špeciálnej syntaxe. Controller riadi tok informácií medzi časťami view a model.

3.2.4 MySQL

MySQL je populárny databázový systém pre webové servery [7]. Tento systém je úspešný hlavne kvôli jeho výkonnosti a rýchlosti. Dokáže bežať aj na minimálnom hardvéri, pričom takmer nezatažuje systémové zdroje. Ďalšou výhodou tohto databázového systému je škálovateľnosť, čo znamená že dokáže rásť spolu s vytvorenou aplikáciou [10].

3.2.5 uWSGI

Projekt uWSGI [8] sa zameriava na vývoj komplexných riešení pre budovanie hostingových služieb. Okrem iných protokolov podporuje aj WSGI (Web Server Gateway Interface), ktorý špecifikuje rozhranie medzi webovým serverom a frameworkom v jazyku Python. Správa dynamického obsahu prebieha komunikáciou medzi webovým serverom (nginx) a django-uwsgi procesom. Na obrázku 3.1 je znázornená komunikácia pri posielaní klientskej požiadavky na webový server.



Obr. 3.1: Komunikácia medzi klientom a serverom pri posielaní klientskej požiadavky. 1.: požiadavka od klienta na webový server, 2.: dynamická požiadavka sa pošle ďalej na uWSGI, 3.: pomocou protokolu WSGI aplikácia uWSGI spustí Django aplikáciu, 4.: aplikácia prístupuje k databáze, 5.: aplikácia prístupuje k statickým súborom.

3.2.6 Google API

Google API je séria API², ktoré umožňujú komunikáciu so službami, ktoré poskytuje Google a integrovanie týchto služieb do iných systémov. Google API umožňuje rozširovanie existujúcej funkcionality iných systémov napríklad použitím vyhľadávania, máp, kalendára a mnohých iných. Okrem rozširujúcich funkcií poskytuje aj autentifikáciu užívateľov, čím poskytuje aplikácii prístup k základným informáciám o užívateľovi a zároveň urýchljuje proces prihlasovania.

²Application programming interface – rozhranie pre tvorbu aplikácií.

Kapitola 4

Návrh aplikácie na sledovanie postupu

Ako bolo spomenuté v predchádzajúcich kapitolách, návrh webovej aplikácie je proces ktorý pozostáva z viacerých fáz. Pred samotnou tvorbou grafického rozhrania je potrebné ujasniť si, akej skupine užívateľov bude aplikácia slúžiť a akú funkcionality by mala poskytovať. Na základe získaných informácií vznikol návrh grafického rozhrania, ktorý bude predstavený v poslednej časti tejto kapitoly.

4.1 Cieľová skupina

Aplikácia je určená pre širokú skupinu užívateľov. Jej využitie sa predpokladá hlavne v rámci skupiny ľudí, ktorí pracujú na rôznych súčiastiach jedného projektu v rámci tímu, ktorý chce dosiahnuť spoločný cieľ. Ďalšie využitie sa predpokladá v rámci skupiny ľudí pracujúcich na podobných projektoch s rovnakým záujmom.

Reprezentatívne príklady užívateľov, na ktoré systém cieľi:

Skupina štyroch študentov, ktorý riešia spoločný školský projekt – spoločné projekty riešia tak na spoločných stretnutiach, ako aj samostatnou prácou, prípadne prácou vo dvojiciach. Preto potrebujú jednoduchý prostriedok, v ktorom sa rýchlo zorientujú a kde všetci vizuálnevidia, na čom projekt stojí, čo by sa mohlo ukázať ako prípadný problém.

Tím malej firmy pracujúci na produkte – udržiavajú prehľad o pokrokoch na projekte, ktorý je zobrazený jednoducho a vizuálne. Na sledovanie chýb a presných úloh používajú nástroj ako Jira, Trello, či Asana. Na sledovanie pokroku a získanie „dashboardu“ využívajú navrhovaný nástroj Gaugeboard.

Malý tím v rámci veľkej firmy – používa nástroj podobne ako v predchádzajúcom bode. Firma má centrálny systém na riadenie projektov, ale Gaugeboard umožňuje získať rýchly vizuálny prehľad o projekte a zároveň ho napríklad bez administratívnych prekážok jednoducho nazdieľať externému zákazníkovi.

Vedúci študentov doktorského alebo magisterského štúdia – potrebuje získavať pravidelné informácie o pokrokoch a výsledkoch svojich študentov. S využitím aplikácie Gaugeboard by mal možnosť jednoducho nazdieľať board medzi študentov a sledovať tak ich prácu aj mimo konzultácií.

4.2 Požiadavky na výsledný systém

Výsledný systém by mal užívateľom umožňovať predovšetkým komunikáciu a referovanie stavu na nimi vybranej záležitosti. To znamená, že základnou požiadavkou na systém je vizuálne zobrazenie miery splnenej práce. S tým súvisí aj možnosť intuitívne nastavovať prípadný pokrok alebo pokles a poskytnúť priestor na krátky komentár súvisiaci s touto zmenou.

Užívateľ by mal mať možnosť vytvárať si pracovné plochy na základe zamerania jednotlivých záležitostí, u ktorých bude sledovať vývoj. Pri sledovanej záležitosti by malo byť možné nastaviť a editovať názov, a umožniť rozdelenie na menšie celky. U konkrétnych častí danej záležitosti by užívateľ mal mať možnosť nastavovať pokrok, pridávať a editovať názov a vizuálne zobrazovanie.

Dôležitou požiadavkou na výslednú aplikáciu je jednoduchý spôsob zdieľania informácií. Aplikácia by mala umožňovať autentifikáciu, zároveň by však mala poskytovať funkcionality aj neprihláseným užívateľom. S autentifikáciou súvisia aj práva jednotlivých užívateľov k úpravám.

Na základe požiadaviek na systém a po premyslení základného návrhu systému som navrhla nasledujúce základné abstrakcie, s ktorými užívateľ (zároveň aj systém) pracuje:

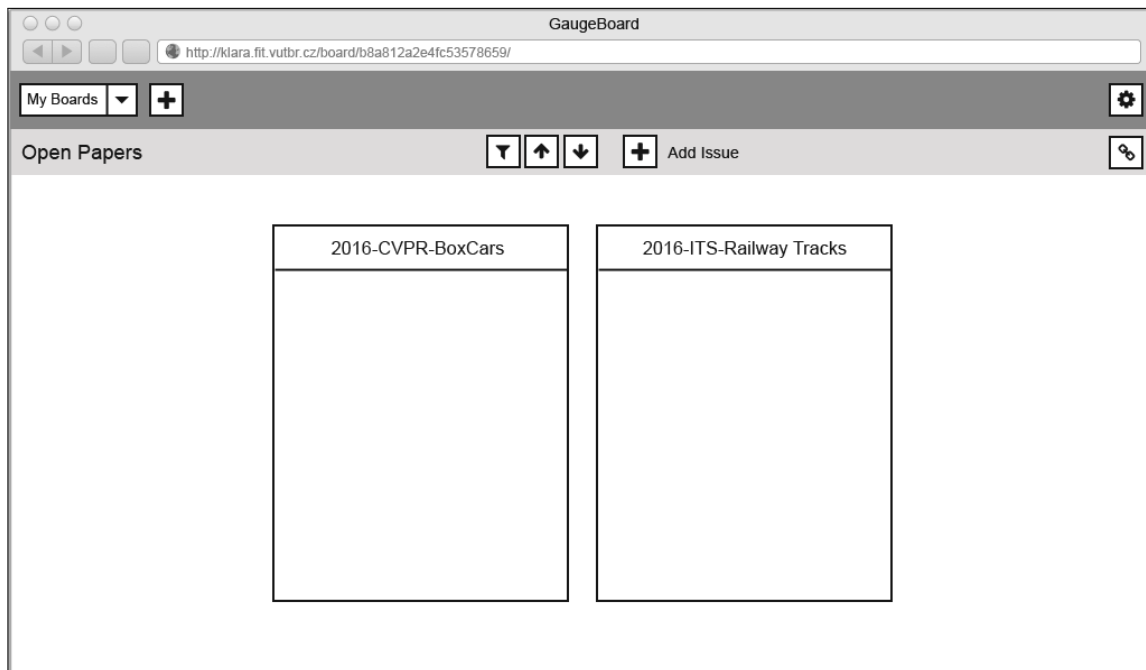
Board – samostatná nástenka zdieľaná tímom. Predstavuje celkový pohľad na projekt, alebo rovnakú oblasť záujmu. Aby bola dostatočne prehľadná, mala by sa vojsť na jednu obrazovku. Ak je projekt zložitejší, môže byť rozdelený do niekoľkých násteniek (gaugeboards). Pokiaľ tím rieši niekoľko rôznych projektov, pre každý vytvorí jednu, prípadne viac násteniek. Ako bolo spomenuté vyššie v kap. 4.1, pri používaní aplikácie učiteľom by mohla byť vytvorená nástenka pre prácu doktorandov, pre študentov magisterského štúdia, pre rozpísané články a pre výučbu. Nástenka by zároveň poskytovala priestor pre porovnávanie výsledkov jednotlivých študentov medzi sebou.

Issue – na nástenke sa vyskytuje niekoľko *záležitostí* (issues). Tento termín je zámerne čo najviac obecný, aby umožňoval najrôznejšie použitie a nesugeroval žiadne zbytočné obmedzenia. Pod týmto pojmom je možné predstaviť si konkrétnu osobu (napríklad študenta spolupracujúceho s učiteľom, ktorý je vlastníkom boardu), jeden rozpísaný článok, jedna časť riešeného softwarového projektu alebo jeden zo športov, ktorými sa užívateľ (športovec zaoberá). Tieto záležitosti môže užívateľ spracovávať sám, alebo má možnosť priradiť editačné práva na každú jednu záležitosť akejkoľvek inej osobe.

Gauge – v rámci jednej záležitosti je niekoľko *ukazateľov* či *meradiel*. Ukazateľ zobrazuje jednu skalárnu veličinu, vyjadrujúcu „naplnenie“ danej kvality, z čoho vyplýva, že je možné uvažovať, že nadobúda hodnoty od 0% do 100%. Vo väčšine prípadov vyjadruje mieru, do akej je *hotová* nejaká práca. Aplikácia je navrhnutá tak, že zámerne nekomunikuje o „hotovej alebo nedokončenej“ úlohe, pretože tento ukazateľ môže vyjadrovať aj subjektívnu mieru spokojnosti, mieru „ako dobre riešiteľ postupoval“, apod.

4.3 Návrh užívateľského rozhrania

Návrh predpokladá realizáciu výslednej aplikácie ako jednu stránku doplnenú dialógovými oknami. Na obrázku 4.1 je zobrazený návrh základnej obrazovky, ktorá je tvorená pracovnou



Obr. 4.1: Základná obrazovka, ktorá zároveň predstavuje tzv. board, čiže pracovnú plochu, ktorá zoskupuje užívateľom vytvorené záležitosti.

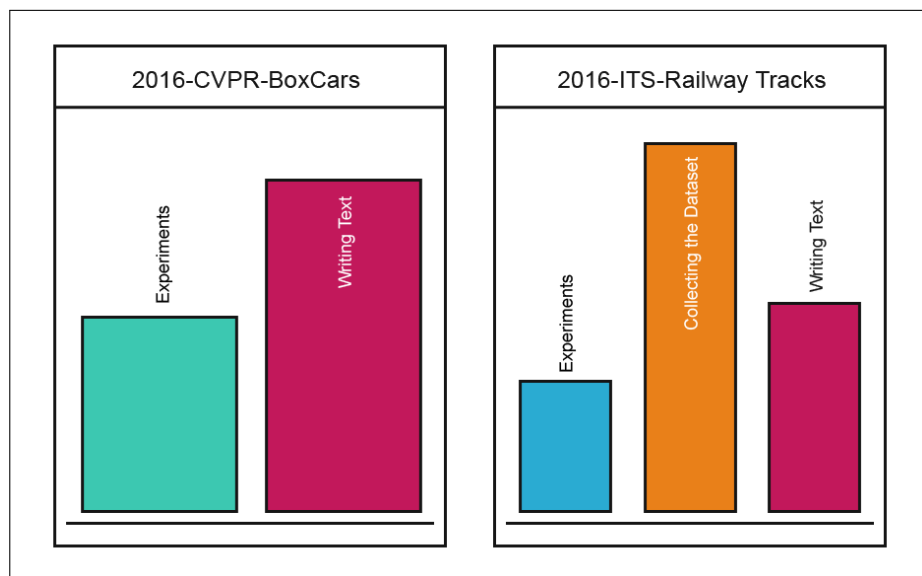
plochou (board). Základná obrazovka sa skladá z hlavného panelu, ktorý obsahuje ovládacie prvky pre vytváranie nových boardov a nastavenia aplikácie. Pre jednotlivé boardy je v návrhu vytvorený ďalší panel, v ktorom sa zobrazuje názov boardu, tlačidlo umožňujúce pridávanie novej issue a tlačidlo na zdieľanie. Tlačidlo na zdieľanie poskytuje užívateľovi URL adresu konkrétneho boardu, prostredníctvom ktorej môže tento board zdieľať s inými užívateľmi bez nutnosti prihlásenia.

Návrh zobrazenia konkrétnej záležitosti a jej súčastí je zobrazený na obrázku 4.2. Z návrhu je evidentné, že každá záležitosť sa skladá z názvu a obsahu. Obsah je rozdelený do niekoľkých stĺpcov (gauges), ktoré si užívateľ vytvorí a pomenuje na základe svojich potrieb.

Na zobrazenie detailnejších informácií o konkrétnej issue slúži dialógové okno. Primárny účel tohto dialógového okna je nastavenie miery pokroku na vybraných úlohách. Dialógové okno je rozdelené do dvoch častí. V ľavej časti sa nachádza vizuálne zobrazenie stavu pokroku na issue. V pravej časti sú vo výpise zobrazené komentáre, ktoré podávajú informáciu o vykonaných zmenách.

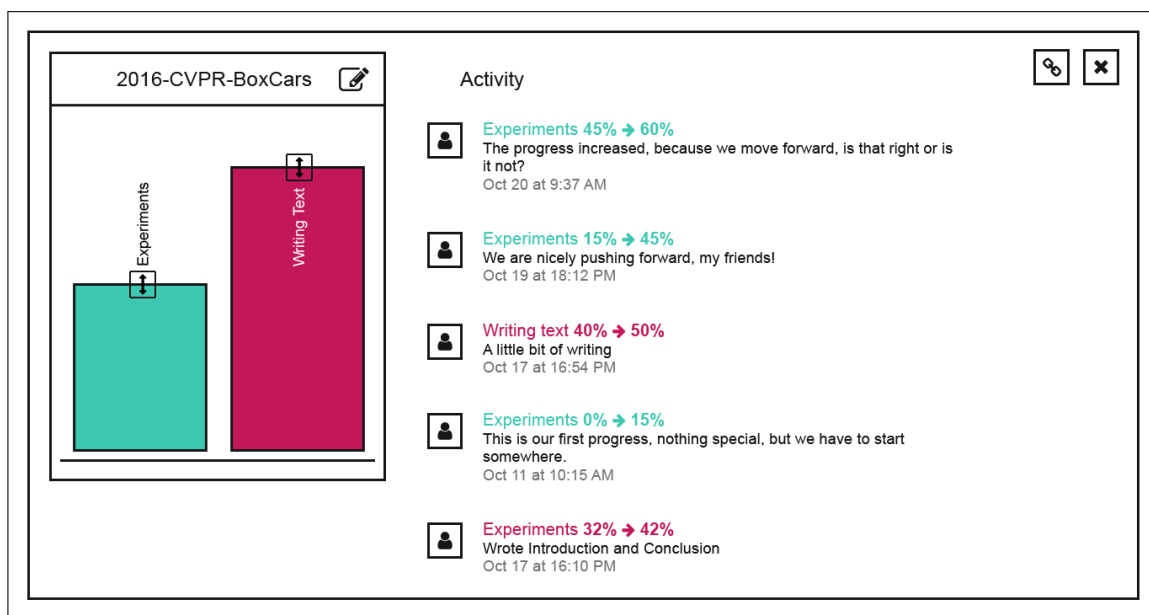
Na obrázku 4.3 je zobrazený detail issue, ku ktorej má užívateľ právo zápisu. Tieto práva má užívateľ automaticky priradené pokiaľ je vlastníkom celého boardu, alebo pokiaľ má prístup k URL adrese konkrétnej issue. Právo na zápis užívateľovi umožňuje manipulovať s hodnotou stavu issue ťahaním stĺpca (gauge) do požadovanej výšky, k čomu ho navádza ikona ťahania. Každú takúto zmenu je povinný popísať stručným komentárom, ktoré sú viditeľné v pravej časti okna. V návrhu je zobrazené editačné tlačidlo, ktoré umožňuje užívateľom manipulovať s nastavením zobrazenia a obsahu issue. Táto funkčnosť je však dostupná len pre prihlásených užívateľov.

V prípade, že užívateľ má editačné práva na vybranú issue v rámci boardu, na ostatné issue má automaticky nastavené práva na čítanie. Práva na čítanie všetkých issues má užívateľ, ktorý má prístup k URL na zdieľanie celého boardu. Po rozkliknutí niektorej

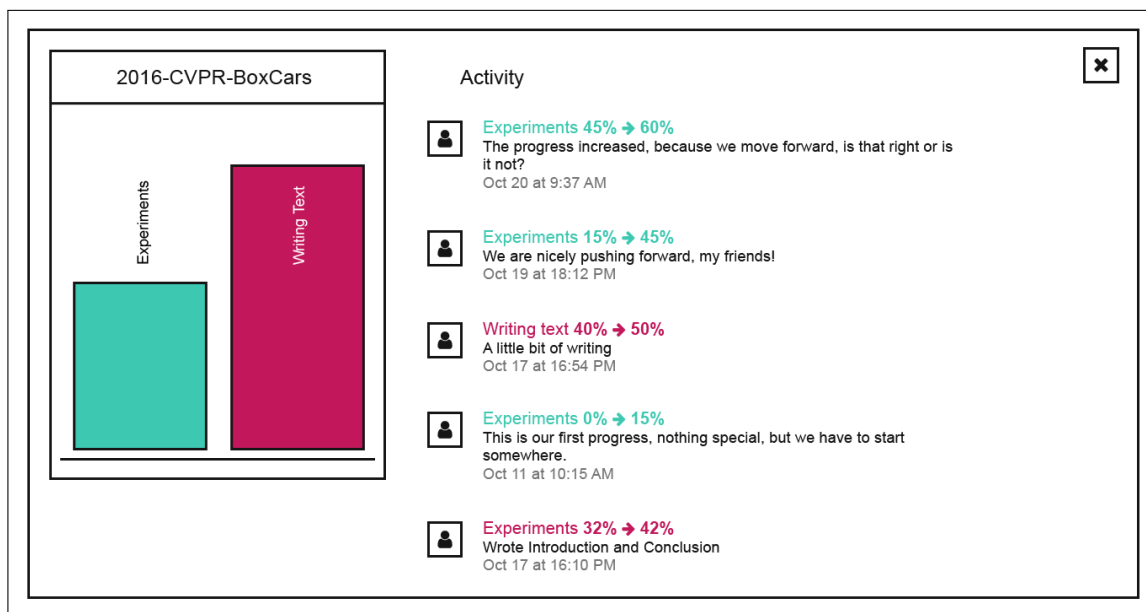


Obr. 4.2: Návrh zobrazenia konkrétnej záležitosti a jej súčastí.

z issues je zobrazený detail issue v takej podobe, ako je možné vidieť na obrázku 4.4. Rozloženie prvkov je veľmi podobné ako pri právach na zápis s tým rozdielom, že užívateľ nemá možnosť manipulovať s hodnotou stĺpca (gauge), nemá možnosť upravovať nastavenia súvisiace so zobrazením a obsahom issue a nemá prístup k URL na jej zdieľanie.



Obr. 4.3: Detail issue s právom na zápis. V ľavej časti sa nachádza vizuálne zobrazenie postupu na issue, ktoré je možné upravovať prostredníctvom ťahania za ikonku. V pravej časti sa zobrazujú komentáre, ktorými užívateľ popisoval vykonané zmeny. Návrh obsahuje editačné tlačidlo s možnosťou nastavenia zobrazovania a obsahu issue a tlačidlo na zdieľanie.



Obr. 4.4: Detail issue s právom na čítanie. V ľavej časti sa nachádza vizuálne zobrazenie postupu na issue. V pravej časti je výpis užívateľských komentárov, ktoré popisujú vykonané zmeny.

2016-CVPR-BoxCars

Issue Title:

Gauges:

Title:

Color:

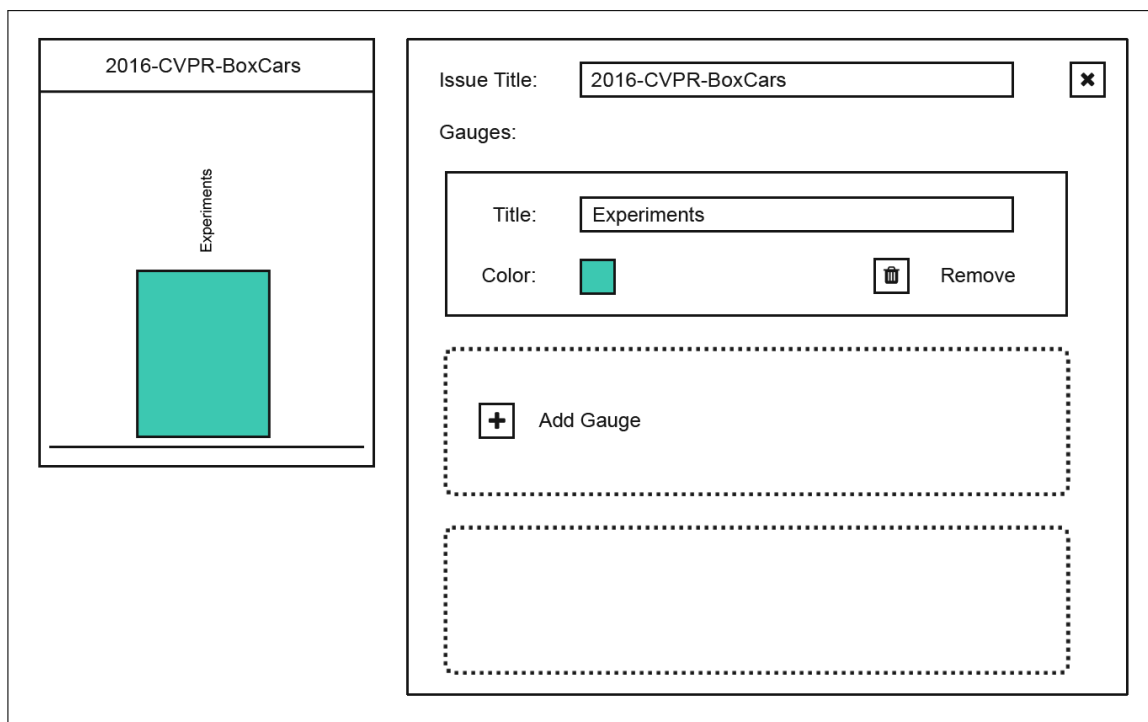
Title:

Color:

Add Gauge

Obr. 4.5: Issue editor, umožňuje nastaviť zobrazenie a obsah jednotlivých issue. Obsahuje ovládacie prvky na zmenu farby, odstránenie a pridávanie gauges a zmenu názvov jednotlivých elementov.

Detail issue obsahuje editačné tlačidlo, po ktorého rozkliknutí sa zobrazí dialógové okno, ktorého návrh je zobrazený na obrázkoch 4.5 a 4.6. Toto okno je takisto rozdelené na dve časti. V pravej časti má prihlásený užívateľ, ktorý má právo manipulovať s issue, možnosť nastaviť obsah a vzhľad konkrétnej issue. V rámci týchto zmien má možnosť meniť názov issue, pridať alebo odobrať gauge a nastaviť im názov alebo farbu. Tieto zmeny sú dynamicky premietané do náhľadu issue v ľavej časti. Ako je vidieť na obrázku 4.6, počet gauge nie je limitovaný konkrétnou hodnotou, návrh počíta s prispôbením zobrazenia na základe potrieb užívateľa.



Obr. 4.6: Issue editor na ktorom je zobrazený spôsob vykresľovania jedného gauge, z ktorého vyplýva, že návrh počíta s premenlivým počtom gauges.

4.4 Návrh serverovej časti

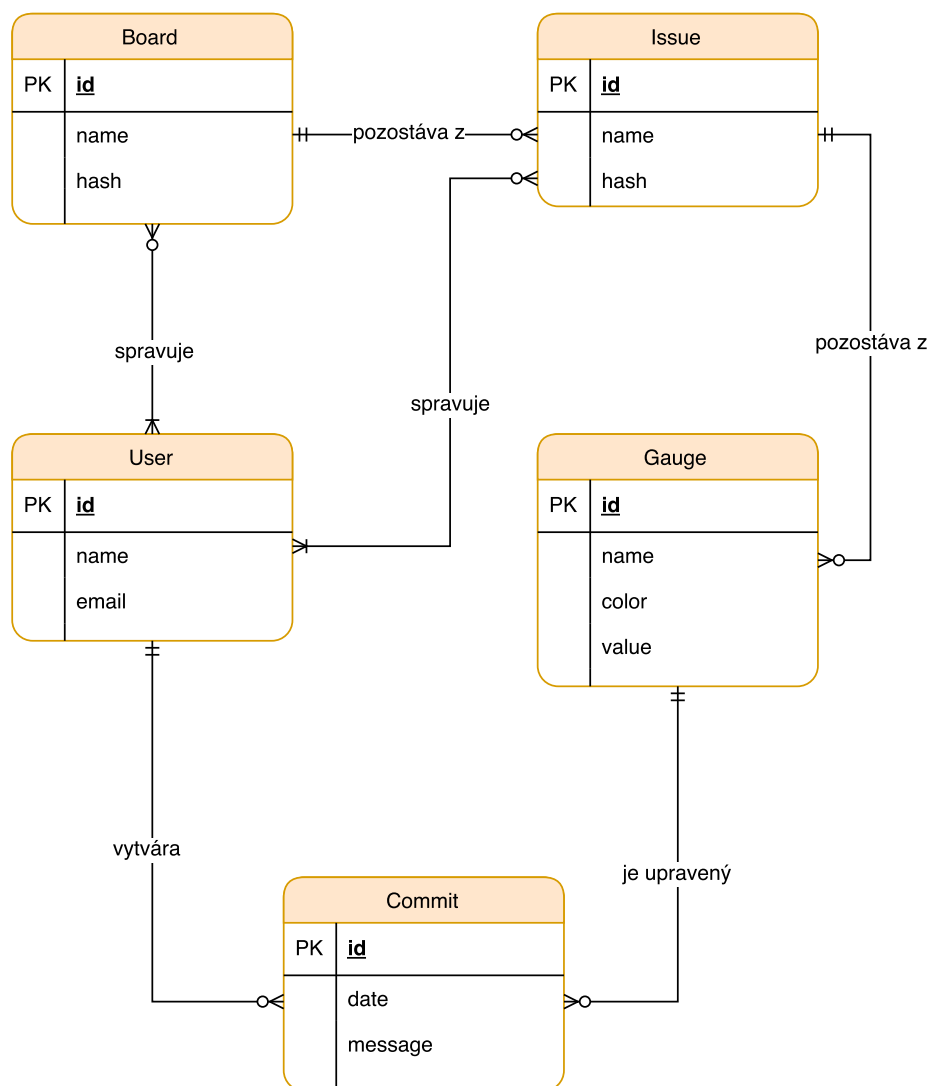
Popri návrhu grafického rozhrania bolo potrebné vytvoriť aj štruktúru výsledného systému. Keďže pre implementáciu aplikácie sa predpokladá využitie webového frameworku Django, systém je automaticky rozdelený do logických celkov model, view a controller. Keďže view predstavuje prezentačnú vrstvu modelu a generuje jednotlivé HTML dokumenty, bola na základe predchádzajúcich návrhov užívateľského rozhrania táto vrstva rozdelená na:

úvodnú stránku – bude poskytovať rozcestník prihláseným užívateľom,

prihlasovaciu stránku – umožňuje užívateľom prihlásiť sa prostredníctvom Google účtu,

stránku pre board – slúži na zobrazovanie všetkých informácií konkrétneho boardu.

Keďže ďalšou časťou výslednej aplikácie je model, ktorý reprezentuje databázové tabuľky relačnej databázy, bola navrhnutá štruktúra týchto tabuliek, ktorá je zobrazená na



Obr. 4.7: Navrhnutý ER diagram.

obrázku 4.7. Na základe požiadaviek na aplikáciu, bol vytvorený návrh dátového modelu, ktorého súčasťou sú entity *board*, *issue* a *gauge*, ktoré boli spomenuté v predchádzajúcej kapitole. Entita *Board*, podobne ako bolo spomenuté v návrhu slúži na uchovanie informácií o celej nástenke. Uchováva užívateľom pridaný názov a vygenerovaný hash, ktorý tvorí jedinečnú časť URL pri zdieľaní. Pozostáva z viacerých issues, ktoré sú reprezentované entitou *issue* a je spravovaný užívateľom, ktorý je reprezentovaný entitou *User*. Entita *Issue* uchováva informácie o užívateľom vytvorených záležitostiach, konkrétne jej názov a vygenerovaný hash, ktorý takisto ako pri entite *board* tvorí časť URL pri zdieľaní. Je spravovaná užívateľom, a pozostáva z gauges, ktoré sú zahrnuté v entite *Gauge*. Táto entita uchováva informácie o jednotlivých ukazateľoch ako ich názov, farbu zobrazenia a užívateľom nastavenú hodnotu. Entita *Commit* reprezentuje užívateľskú správu popisujúcu vykonanú zmenu, obsahuje dátum zmeny a text tejto správy.

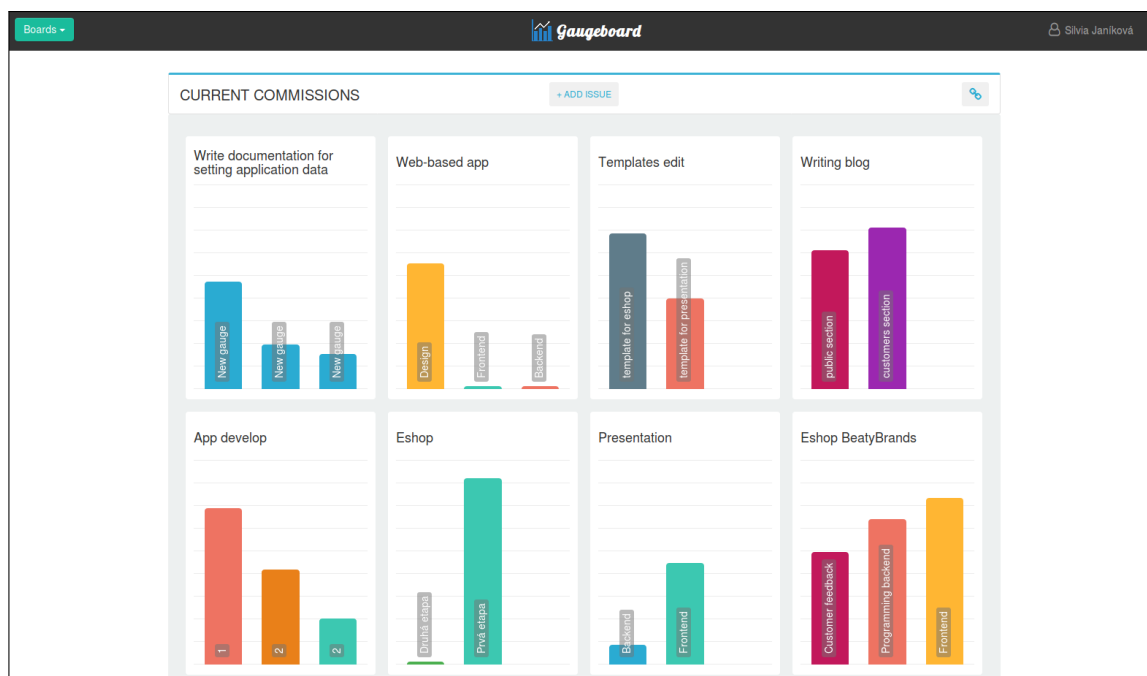
Kapitola 5

Implementácia

V tejto kapitole bude popísaný spôsob implementácie výslednej aplikácie. Je tu opísaná tvorba grafického rozhrania, a komunikácia medzi klientskou a serverovou časťou. V kapitole je ďalej vysvetlený postup pri implementácii zdieľania prvkov systému a autentifikácia, ktorá bola realizovaná využitím Google API.

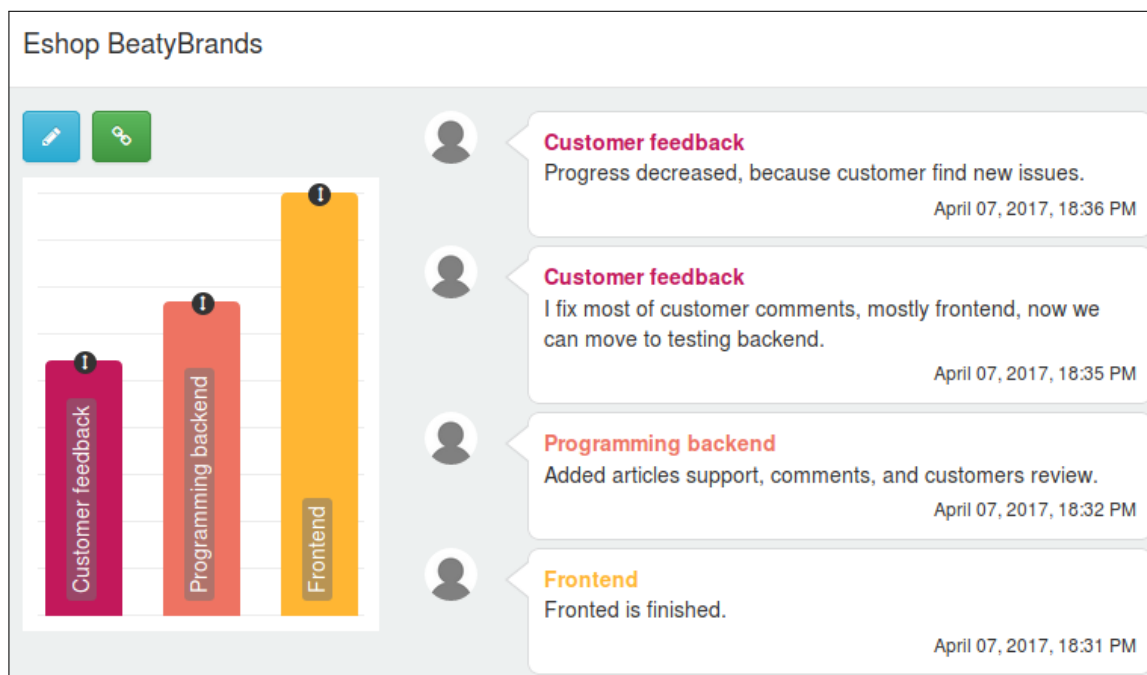
5.1 Grafické rozhranie systému

Na dynamické generovanie HTML obsahu využíva Django systém šablón. Implementácia grafického rozhrania výslednej aplikácie je teda rozdelená do niekoľkých súborov. Obsahuje samostatný súbor pre základnú obrazovku, ako je možné vidieť na obrázku 5.1. Pri implementácii tejto obrazovky sa postupovalo s ohľadom na budúcu potrebu responzívnej verzie, preto bolo od začiatku potrebné využívať triedy z Bootstrapu.



Obr. 5.1: Základná obrazovka aplikácie, zobrazuje výpis užívateľskej nástenky (boardu).

Pri vytváraní hlavného obsahu (šedý blok) bola využitá trieda *container*. Pri plávajúcich prvkoch boli použité špeciálne triedy zo systému mriežky (grid system), ktoré takisto poskytuje Bootstrap.



Obr. 5.2: Detail záležitosti (issue).

Pre implementáciu modálneho okna, ktorý obsahuje detail záležitosti zobrazený na obrázku 5.2, bola použitá komponenta *modal*, ktorú poskytuje Bootstrap. Jednotlivé prvky sú rozložené využitím systému mriežky. Pri implementácii posuvného meradla (gauge), ako je možné vidieť v ľavej časti toho istého obrázku bola použitá funkcia *resizable*, ktorá je dostupná v knižnici jQueryUI. Táto funkcia umožňuje meniť veľkosť jednotlivých elementov ťahaním za okraj elementu, prípadne za úchopovú ikonu.

Zobrazovanie aplikácie na zariadeniach a rôznou šírkou displeja, ako je vidieť na obrázku 5.3, čiže responzívne zobrazovanie, čiastočne zaručuje využitie pripravených tried z Bootstrapu. Avšak pre celkový výsledok bolo potrebné do hlavičky dokumentu pridať:

```
<meta name="viewport" content="width=device-width, initial-scale=1.0">
```

Viewport poskytuje prehliadaču informácie o tom ako stránku zobrazí stránku so správnou mierkou (*initial-scale*) a rozmerom (*device-width*). Okrem prispôbenia veľkosti zobrazenia bolo potrebné upraviť aj niektoré špecifické prvky ako obsah horného panelu, alebo tlačidlo na pridávanie nových prvkov.

5.2 Komunikácia medzi klientskou a serverovou časťou

Na základe návrhu užívateľského rozhrania bude užívateľ pristupovať k dátam v rámci jednej stránky. Za týmto účelom bolo navrhnuté API, pomocou ktorého užívateľ k týmto dátam pristupuje, upravuje ich alebo vytvára nové. Klient môže posilať na server asynchrónne HTTP požiadavky na rôzne URL podľa toho, ku ktorej entite chce pristupovať:



Obr. 5.3: Zobrazenie základnej obrazovky pri zariadeniach s menšou šírkou displeja.

- *api/v1/dashboards*,
- *api/v1/issue*,
- *api/v1/gauge*,
- popr. na zaslanie zdieľacieho odkazu *api/v1/sharing*.

Na strane klienta je API implementované pomocou ajaxových požiadaviek, ktoré sú na strane serveru spracované a vyhodnotené pomocou niekoľkých funkcií. Tieto funkcie umožňujú dynamicky premietiť užívateľom vykonané zmeny. Funkcie *issues* a *scales* poskytujú vytvorenie, vymazanie a zmenu pomenovania prvkov *issue* a *gauge* a zmenu farby prvku *gauge*. V rámci vytvoreného API je implementovaná funkcia *sharing*, ktorej úlohou je v odpovedi poslať hash na požadovaný prvok.

5.3 Nasadenie aplikácie na server

Aplikácia bola nasadená na školský server *klara.fit.vutbr.cz*. Pre implementáciu bolo použité virtuálne prostredie, ktoré sa používa pre efektívny vývoj aplikácií. Vďaka tomuto mechanizmu má každá aplikácia potrebné závislosti a nastavenia oddelené od iných. Virtuálne prostredie bolo vytvorené pomocou príkazu `python3 -m venv /path/to/newvirtualenv`. V tomto prostredí sa nainštaloval framework Django s potrebnými závislosťami.

Ďalším krokom bola konfigurácia webového serveru nginx v konfiguračnom súbore, ktorý sa nachádza v zložke `/etc/nginx/sites-enabled/`. V tomto súbore je potrebné nastaviť názov hostiteľa, port a relatívnu cestu, cez ktorú bude pristupovať klient k aplikácii. Taktiež sa tu uvádza cesta k statickým súborom a cesta k soketu, cez ktorý bude server komunikovať s rozhraním uWSGI.

Pri konfigurácii uWSGI bolo nutné vytvoriť konfiguračný súbor, ktorý obsahuje cesty k Django aplikácii, ku konkrétnemu modulu, ktorý sa stará o WSGI komunikáciu a cestu k virtuálnemu prostrediu aplikácie. Súčasťou tohto súboru je ďalej cesta k soketu cez ktorý komunikuje s webovým serverom.

5.4 Zdieľanie prostredníctvom URL

Pre zdieľanie konkrétnej issue, alebo konkrétneho boardu s inými užívateľmi sa používa špecifická URL. Jedinečnosť tejto URL adresy zabezpečuje hašovací funkcia *SHA256*, ktorá je naimplementovaná v module *hashlib*. Vstupom tejto funkcie je unikátny identifikátor boardu, prípadne unikátny identifikátor boardu aj konkrétnej issue. Vo funkcii je táto hodnota prevedená na hexadecimálny reťazec, ktorý je orezaný na dĺžku štrnástich znakov pokiaľ sa jedná o celkový board, alebo na dĺžku pätnástich znakov v prípade generovania hashu pre samostatnú issue. Tento reťazec je uložený do prislúchajúcej tabuľky v databáze. Pri zdieľaní vybranej časti aplikácie sa vygeneruje modálne okno s URL adresou, pričom táto adresa sa pomocou aplikačného rozhrania získava až po stlačení tlačidla na zdieľanie.

5.5 Autentifikácia pomocou Google API

Užívatelia, ktorí pracujú s aplikáciou nemusia byť autentifikovaní. Neprihlásený užívatelia sú však limitovaný v niektorých funkcionalitách ako vytváranie nového boardu, zmena poradia prvkov, apod. Užívatelia majú možnosť autentifikovať sa pomocou google účtu, čo zabezpečuje Google API. Táto autentifikácia prebieha pomocou protokolu OAuth 2. Na využitie autentifikácie od inej služby bolo potrebné v Django zahrnúť modul *django-social-auth* [6].

Pre správne fungovanie autentifikácie bolo potrebné nakonfigurovať niektoré zo zdrojových súborov. V súbore *settings.py* bolo potrebné pridať privátne kľúče, ktoré je možné vygenerovať v Google účte pre OAuth 2 autentifikáciu: Tieto privátne kľúče sa vkladajú do premenných *SOCIAL_AUTH_GOOGLE_OAUTH2_KEY* a *SOCIAL_AUTH_GOOGLE_OAUTH2_SECRET*.

V rovnakom súbore je nutné pridať aplikáciu *social_django* do zoznamu *INSTALLED_APPS*. *Social_django* je modul pre Python, ktorý sa po pridaní do tohto zoznamu stáva povoleným modulom do Django. V zozname *AUTHENTICATION_BACKENDS* je potrebné pridať modul *social_core.backends.google.GoogleOAuth2*, ktorý sa stará o autentifikáciu užívateľa pomocou konkrétnej služby.

Kapitola 6

Testovanie a možnosti ďalšieho rozvoja aplikácie

6.1 Testovanie

Po implementácii funkčného prototypu prebehlo užívateľské testovanie. Testovanie bolo realizované na dvoch rozdielnych projektoch. Podstatou prvého projektu bolo sledovanie postupu na bakalárskych prácach niekoľkých študentov FITu, z čoho vyplýva, že sa jednalo o skupinu respondentov približne v tom istom veku s dobrou technickou vyspelosťou. Druhý projekt bol zameraný skôr na bežných užívateľov, skupinu respondentov tvorili užívatelia v rôznom veku (20–50 rokov). Testovanie prebiehalo niekoľko dní, počas ktorých respondenti podávali informácie o svojich pokrokoch v rámci svojej skupiny. Po ukončení tohto testovania respondenti vyplnili krátky dotazník, v nasledovnú znení:

- Ako na vás táto aplikácia pôsobí?
- Podarilo sa vám v nej zorientovať?
- Aké nové funkcie by ste privítali?
- Čo vám vyhovuje/nehovuje na vzhľade?

6.2 Výsledky testovania

Na základe odpovedí z dotazníka a sledovania niektorých užívateľov bolo zrejmé, že užívateľom sa veľmi páčilo jednoduché spracovanie užívateľského rozhrania. Ocenili prehľadné zobrazenie všetkých potrebných informácií, bez nutnosti hlbšieho skúmania fungovania aplikácie. Z čoho vyplýva, že sa im aplikácii dobre orientovalo, s výnimkou skupinky starších respondentov, ktorým robilo mierne problémy nastavenie pokroku (potiahnutie za konkrétny stĺpec). Užívatelia sa vyjadrovali aj k funkciám, ktoré by od aplikácie očakávali. Z týchto funkcií sa u rôznych užívateľov opakovali:

Radenie issue a gauges – viacerí užívatelia by ocenili funkciu, ktorá by umožňovala zmenu poradia issues v rámci jedného boardu a takisto gauges v rámci jednej issue.

Editovanie správy – v niektorých prípadoch by podľa respondentov bola vhodná funkcia editovania pridanej informácie o stave pokroku. Na základe ich odpovedí by bola užitočná hlavne pri opravovaní preklepov.

Kopírovanie issue – pri vytváraní boardu, ktorý obsahuje niekoľko issues s podobným obsahom je ich naklikávanie pomerne zdĺhavé. Respondenti by uvítali možnosť kopírovania existujúcej issue.

Vzhľad aplikácie hodnotili respondenti dobre. Jeden z respondentov sa vyjadril, že by zvažil umiestnenie ikon na zdieľanie a editovanie issue, ktoré sa nachádzajú v detaile issue. Niektorí z respondentov by zase ocenili vertikálnu osu pri zobrazení gauge, ktorá by označovala percentuálny pokrok.

Všetky pripomienky boli dôkladne vyhodnotené a do aplikácie boli zapracované vybrané z nich. Z funkcionality boli zrealizované niektoré menšie zmeny. V pôvodnej verzii aplikácie boli novo pridané prvky (issue a gauge) zaradené vždy na začiatok bloku. Z testovania vyplynulo, že pre užívateľov je logickejšie, keď sú tieto prvky radené na posledné miesto. Z rozsiahlejších funkcií boli realizované zmena poradia issues a gauges a pridanie kontextového menu. Medzi užívateľskými pripomienkami sa vyskytla aj funkcia na editovanie správy pridávanej pri zmene pokroku. V rámci editovania existujúcej informácie o pokroku by mohol užívateľ úplne zmeniť obsah pôvodnej správy, čo by mohlo viesť k strate skutočnej histórie vykonaných zmien. Z tohto dôvodu som sa rozhodla túto funkciu nerealizovať.

6.2.1 Zmena poradia issue a gauge

Funkcia, ktorá by umožňovala zmenu poradia jednotlivých issues v rámci boardu, alebo gauges v rámci issue by mohla byť z pohľadu užívateľského rozhrania implementovaná ako „drag and drop“. Táto funkcia by mala byť prístupná len užívateľom, ktorý sú vlastníkom celého boardu (v prípade presúvania issues), alebo užívateľom, ktorý majú možnosť editovať issue a ktorý sú zároveň prihlásení (v prípade presúvania gauges). Na realizáciu tejto funkcie na klientskej strane bola využitá funkcia *sortable* z knižnice jQueryUI, ktorá zabezpečuje presúvanie prvkov. Na strane serveru bolo potrebné v prvom rade rozšíriť existujúcu databázovú tabuľku o atribút *position* a vytvoriť príslušné funkcie v API.

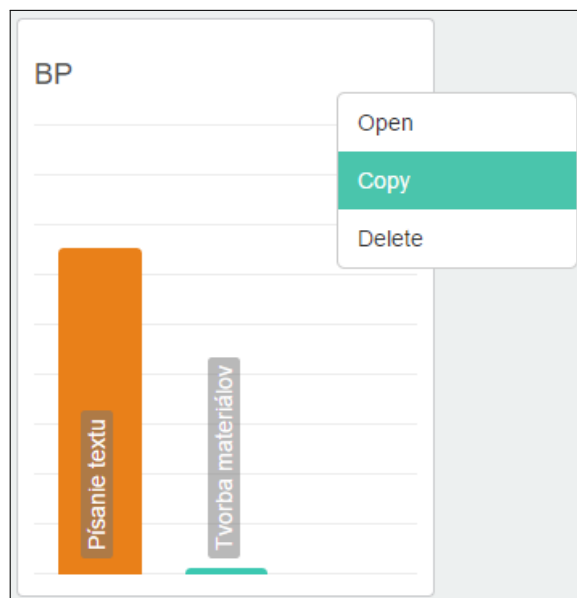
6.2.2 Pridanie kontextového menu

Ďalšou funkciou, ktorú by respondenti očakávali od aplikácie, bola možnosť kopírovania existujúcej issue. Na realizovanie kopírovania bola vytvorená funkcia v aplikačnom rozhraní, ktorá vytvára kópiu databázového objektu spolu so všetkými jej súčasťami (gauges). V prípade kopírovania issue, na ktorej už bol zaznamenaný pokrok sa hodnota gauge spolu s textovou informáciou do novej issue neprenášajú. Po vytvorení duplikovanej issue sa do názvu pridá reťazec „copy:“, aby bolo možné jednotlivé issue od seba odlíšiť. Na základe pridania tejto funkcie bolo vytvorené kontextové menu, ktoré je možné vidieť na obrázku 6.1. Okrem možnosti kopírovania obsahuje zobrazenie detailu issue a odstránenie issue.

6.3 Možnosti ďalšieho rozvoja aplikácie

Existuje niekoľko funkcií, o ktoré by aplikácia mohla byť v budúcnosti obohatená. Prvou z nich je pridávanie príloh do informácie o pokroku. Pri niektorých projektoch by užívatelia mohli využívať na referovanie stavu projektu obrázky, alebo textové dokumenty na ktorých práve pracujú.

Ďalšou možnosťou je vizuálne reflektovanie zmeny. Pri nastavení výšky gauge sa výška meradla znižuje alebo zvyšuje. Momentálne je táto udalosť zaznamenaná pri textovej informácii o vykonanej zmene. V budúcnosti by mohla byť vytvorená funkcia, ktorá by umožňo-



Obr. 6.1: Kontextové menu pre issue, ktoré bolo vytvorené na základe pridania možnosti kopírovania issue.

vala grafické zobrazenie predchádzajúcej hodnoty konkrétneho meradla. Táto informácia by mohla byť zobrazená napríklad pri kliknutí na prislúchajúcu textovú informáciu o vykonanej zmene.

Aplikácia by mohla byť rozšírená aj o zasielanie notifikácií na email. Pre výzvu užívateľov k referovaniu stavu na ich projekte by mohla aplikácia poskytovať možnosť zasielať emailové notifikácie. Pri každej issue by bola možnosť vyplnenia notifikačného emailu, do kópie by sa vložil email správcu boardu a konkrétny text by si správca doplnil podľa potreby.

Kapitola 7

Záver

V tejto práci bol opísaný proces tvorby webovej aplikácie zameranej na sledovanie postupu práce v rôznych oblastiach. Táto aplikácia umožňuje prehľadné sledovanie pokroku v rámci skupiny užívateľov, čím vzniká priestor na porovnávanie dosiahnutých výsledkov a tým je užívateľ motivovaný k vyššej produktivite.

V jednotlivých kapitolách bol predstavený proces tvorby návrhu užívateľského rozhrania a nástroje použité k implementácii. Návrh užívateľského rozhrania odpovedá konceptu jednostránkovej aplikácie, ktorý má v súčasnosti veľký potenciál. Cieľom tohto projektu bolo experimentovať s úplne novým prístupom k organizácii tímov, ktorý by bol ešte jednoduchší ako tie najjednoduchšie dostupné (Trello), a ktorý by nevedojak poskytoval celkový náhľad na riešený „projekt“ v podobe akéhosi „dashboardu“. Z tohto dôvodu som najprv vytvorila funkčný prototyp, ktorý neumožňoval prihlásenie užívateľov, zdieľanie a pokročilé funkcie, ale slúžil čisto k testovaniu a vylepšovaniu užívateľského rozhrania *boardu*, *issues* a *gauges*. Na základe testovania bola aplikácia rozšírená o niekoľko praktických funkcií. Nad rámec tohto prototypovania som aplikáciu dokončila do podoby, ktorá by mohla byť pripravená do reálnej prevádzky. V súčasnosti je aplikácia funkčná, umožňuje prihlásenie cez Google účet, anonymné (bez nutnosti prihlásenia, na podobnom princípe ako Doodle¹) zdieľanie odkazov pre modifikáciu jednotlivých issues.

Svoju prácu na aplikácii Gaugeboard som prezentovala na študentskej konferencii Excel@FIT.

¹<http://doodle.com/>

Literatúra

- [1] George, N.: The Model-View-Controller Design Pattern. online, [cit. 2017-04-07], <http://djangobook.com/model-view-controller-design-pattern/>.
- [2] Krug, S.: *Don't make me think: A common sence approach to web usability*. New Riders Publishing, 2006, ISBN 0-321-34475-8.
- [3] Lardinois, F.: Atlassian starts integrating more of its products with Trello. *techcrunch.com*, March 2017.
- [4] Asana team: online, [cit. 2017-04-07], <https://asana.com/>.
- [5] Django Software Foundation: online, [cit. 2017-05-11], <https://www.djangoproject.com>.
- [6] Django Software Foundation: online, [cit. 2017-05-11], <https://python-social-auth.readthedocs.io/en/latest/intro.html>.
- [7] Oracle Corporation: online, [cit. 2017-05-11], <https://www.mysql.com/>.
- [8] uWSGI team: online, [cit. 2017-05-11], <https://uwsgi-docs.readthedocs.io/>.
- [9] Methvin, D.; Borchers, K.: online, [cit. 2017-05-11], <https://jquery.com/>.
- [10] Nixon, R.: *Learning PHP, MySQL, JavaScript, and CSS*. O'Reilly Media, Inc, 2012, ISBN 978-1-4493-1926-7.
- [11] Otto, M.; Thornton, J.: online, [cit. 2017-05-11], <http://getbootstrap.com/>.
- [12] RingCentral: online, [cit. 2017-05-03], <https://glip.com/>.
- [13] Tarasowski, M.: How to build the next Trello and sell it for 425 million or more. online, [cit. 2017-04-07], <https://disruptnext.com/how-to-build-the-next-trello-and-sell-it-for-425-million-or-more-589045c9bd64>.
- [14] Řezáč, J.: *Web ostrý jako břitva: návrh fungujícího webu pro webdesignery a zadavatele projektů*. House of Řezáč, 2016, ISBN 978-80-270-0644-1.

Prílohy

Príloha A

Obsah priloženého CD

src Zložka so zdrojovými súbormi aplikácie.

src_ts Zložka so zdrojovými súbormi technickej správy.

technicka_sprava.pdf Text práce.

plagat.pdf Prezentačný plagát.

video.mp4 Demonštračné video.